



netidee

PROJEKTE

CoMatrix

Endbericht | Call 13 | Projekt ID 3441

Lizenz CC-BY-SA

Inhalt

1	Einleitung.....	3
2	Projektbeschreibung.....	3
2.1	CoMatrix Zielgruppen.....	4
2.2	Projektergebnisse	5
3	Verlauf der Arbeitspakete.....	5
3.1	Arbeitspaket 1 - <i>Projektstart</i>	5
3.2	Arbeitspaket 2 – <i>IoT Protokolle</i>	6
3.3	Arbeitspaket 3 – <i>Matrix Infrastruktur</i>	7
3.4	Arbeitspaket 4 – <i>Matrix for Constrained Devices</i>	7
3.5	Arbeitspaket 5 – <i>Öffentlichkeitsarbeit</i>	8
3.6	Arbeitspaket 6 – <i>Wissenschaftliche Verarbeitung</i>	9
3.7	Arbeitspaket 7 – <i>Projektmanagement</i>	9
3.8	Arbeitspaket N – <i>Dokumentation und Formales am Projektende</i>	9
4	Umsetzung Förderauflagen	10
5	Liste Projektergebnisse	10
6	Verwertung der Projektergebnisse in der Praxis	10
7	Öffentlichkeitsarbeit/ Vernetzung.....	11
8	Eigene Projektwebsite.....	11
9	Geplante Aktivitäten nach netidee-Projektende.....	11
10	Anregungen für Weiterentwicklungen durch Dritte.....	12

1 Einleitung

Das Matrix Protokoll ist dabei, den Messaging Markt zu revolutionieren. Wir halten einen ähnlichen Impact dieses Protokolls auch im IoT Bereich für möglich und möchten dazu beitragen.

Matrix setzt auf eine klassische TLS/HTTP/REST/JSON Architektur. Das Ziel war es, ein Gateway zu erstellen, das diese Kommunikationssemantik auf den IETF Constrained Stack DTLS/CoAP/CBOR herunterbricht und auf der einen Seite mit den Constrained-IoT-Devices kommuniziert, auf der anderen Seite die Kommunikation wieder in den Standard Stack (die Matrix Client-Server API) umsetzt und so eine einheitliche Kommunikation ermöglicht. Dazu wurde eine Adaption des Matrix Protokolls und dessen Implementierung für Constrained-Devices durchgeführt; somit können neue Zielgruppen im IoT Bereich erschlossen werden. Es wurden Szenarien, die IoT Geräte über das Matrix Protokoll kommunizieren lassen, konzipiert, implementiert und vollständig dokumentiert, um zukünftige Entwicklungen zu erleichtern.

Die Arbeiten am Projekt CoMatrix wurden ab Mitte September 2020 wieder aufgenommen bzw. neugestartet, nachdem es durch personelle Veränderungen zu erheblichen Verzögerungen kam.

Der Projektleiter war Matthias Hudler, der auch den Studiengang IT-Security an der FH-Campus Wien leitet und dem Kompetenzzentrum für IT-Security vorsteht. Verstärkt wurde das Team durch Tobias Buchberger und Ines Kramer, beide haben an der FH-Campus Wien den Masterstudiengang IT-Security abgeschlossen und sind derzeit dort als wissenschaftliche Mitarbeiter*innen tätig.

2 Projektbeschreibung

Es gibt zahlreiche Protokolle, die im Internet of Things verwendet werden. Einen wirklich allgemeinen Einsatz scheint keines davon zu erlauben. Entweder sind sie zu schwergewichtig für Constrained-Devices oder -Networks, wie die TCP-basierten http- und MQTT-Protokolle, oder sie bieten aufgrund ihres einfachen Aufbaus gewünschte Features nicht. Ein Protokoll, welches solche Features ermöglicht und gleichzeitig über alle IoT-Geräteklassen hinweg eingesetzt werden kann, könnte den Markt revolutionieren.

Das Matrix Protokoll setzt auf eine klassische TLS/HTTP/REST/JSON Architektur. Das Ziel war es, ein Gateway zu erstellen, das diese Kommunikationssemantik auf den IETF Constrained Stack DTLS/CoAP/CBOR herunterbricht und auf der einen Seite mit den Constrained IoT Devices spricht, auf der anderen Seite die Kommunikation wieder in den Standard Stack (die Matrix Client-Server API) umsetzt und so eine einheitliche Kommunikation ermöglicht. Dazu wurde eine Adaption des Matrix Protokolls und dessen Implementierung für Constrained Devices durchgeführt; somit können neue Zielgruppen im IoT Bereich erschlossen werden. Es wurden Szenarien, die IoT Geräte über das Matrix Protokoll kommunizieren lassen, konzipiert, implementiert und vollständig dokumentiert, um zukünftige Entwicklungen zu erleichtern.

Die Dokumentation von Szenario, Hardware und Software eines CoMatrix-basierten IoT Netzwerkes ermöglicht es Lehrenden, diese rasch und unkompliziert in ihren Unterricht einbauen zu können.

Für den grundlegenden Einsatz von Matrix im IoT Bereich gab es bis dato keine praxistauglichen Projekte (vgl. <https://matrix.org/iot/>). Hierfür leisteten wir strukturierte Aufbauarbeit. Auf Protokollebene will CoMatrix eine Brücke schlagen: Gleichzeitig ressourcenschonende Kommunikation für alle Geräte zu ermöglichen, die diese brauchen, als auch ohne Protokollbruch eine zuverlässige Kommunikation für stärkere Geräte zu garantieren. Etwas Vergleichbares bietet derzeit keines der zahlreichen Internet of Things Protokolle.

Ein universelles Protokoll welches für vielfältige Einsatzzwecke und Hardwaregrößen im IoT geeignet ist, ist sowohl für Software Entwickler*innen als auch für Hardwarehersteller interessant. Ein breiter Einsatz ist global wie lokal in Österreich denkbar. Zusätzlich können Lehrende und Studierende auf Österreichischen Hochschulen von den Ergebnissen profitieren.

Das Projekt wird im Rahmen der Lehr- und Forschungstätigkeit an der FH Campus Wien weitergeführt und wird auch über diese finanziert. Die aus CoMatrix resultierenden Projektergebnisse- und Erkenntnisse werden den Studierenden in Lehrveranstaltungen vermittelt. Eine Weiterentwicklung des Projektes im Rahmen von Projektarbeiten sowie Abschlussarbeiten ist geplant. Daraus ergibt sich eine regelmäßige Aktualisierung der Projektergebnisse.

2.1 CoMatrix Zielgruppen

Folgende Zielgruppen wollen wir erreichen:

- **Matrix/Open-Source Community:** Die Matrix Community besteht federführend aus der Matrix.org Foundation. Die Entwickler bezeichnen das Internet of Things als einen der vier Schwerpunkte für die Nutzung von Matrix. Gleichwohl das IoT eines der Schwerpunkte von Matrix ist, haben sich die bisherigen Entwicklungen hauptsächlich auf den Messaging Bereich fokussiert. Ohne Beispiele und Erfahrungswerte ist ein Einstieg schwierig. CoMatrix fungiert daher als Kickstarter für den IoT Einsatz von Matrix.
- **Organisationen:** Lehrende an allen universitären Bachelor- oder Master-Studiengängen, die sich mit dem Bereich IoT bzw. Embedded Systems beschäftigen. Für einen effizienten Einsatz im Unterricht sind gut aufbereitete Beispiele und Dokumentationen notwendig, die gerade im IoT Bereich derzeit leider sehr rar sind.
- **Unternehmen:** Heimische (oder internationale) Unternehmen, die IoT Lösungen selbst oder für Kunden entwickeln.

2.2 Projektergebnisse

Im Wesentlichen konnten folgende Projektergebnisse erreicht werden:

- „CoMatrix Gateway“: Eine Gateway Serversoftware, die auf einem Border Router laufen kann und auf der einen Seite mit Constrained-Devices über das CoAP basierte CoMatrix Protokoll kommuniziert und auf der anderen Seite für den Austausch mit Matrix Homeservern das reguläre Matrix Protokoll einsetzen kann. In unserer Testumgebung kam ein Raspberry 3B+ als CoMatrix Gateway mit einem OpenLabs Raspberry Pi 802.15.4 Radio Modul zum Einsatz, aber die einzige Anforderung an die Hardware/Betriebssystem ist die Unterstützung von Python3.
- „CoMatrix Library“: Ein Software-Modul für das RIOT-OS (<https://www.riot-os.org/>) und ressourcenschwache IoT Devices, um über das CoMatrix Protokoll mit dem Gateway zu kommunizieren. In unserer Testumgebung kam ein Atmel SAM R21 Xplained Pro Evaluation Kit (SAMR21-xpro) zum Einsatz, aber grundsätzlich sollte die Library auf allen, von RIOT unterstützten und netzwerkfähigen, Boards mit ausreichend Arbeitsspeicher verwendet werden können (vgl. <https://www.riot-os.org/boards.html> *ausgenommen* <https://github.com/RIOT-OS/RIOT/blob/master/examples/gcoap/Makefile.ci>).
- „Dokumentation“: Eine ausführliche User/Developer Dokumentation, wie das (Co)Matrix Protokoll auf allen geeigneten IoT Devices eingesetzt werden kann und wie das CoMatrix Gateway zu verwenden bzw. zu betreiben ist. An dieser Stelle werden auch die Testumgebung und entworfenen IoT Szenarien beschrieben.

3 Verlauf der Arbeitspakete

Von September 2019 bis September 2020 war die Arbeit, aufgrund nicht vorhandener personeller Ressourcen, an diesem Projekt unterbrochen. Alle Arbeitspakete wurden hinsichtlich des ursprünglichen Projektplans verspätet fertiggestellt, da das Projekt neugestartet wurde, nachdem der ursprüngliche Projektleiter Mathias Tausig die FH Campus Wien verlassen hat. Außerdem hat ein studentischer Mitarbeiter (Andreas Gadermaier), der am Projekt mitgearbeitet hat, ebenfalls die FH Campus Wien verlassen. Das Projekt wurde nach Ansuchen des neuen Projektleiters, Matthias Hudler, letztendlich bis Anfang Mai 2021 verlängert.

3.1 Arbeitspaket 1 - *Projektstart*

AP-Tätigkeiten wurden durchgeführt laut Beschreibung (Vertragsprüfung, Projektplanung, Blog-Erstellung).

Der Vertrag wurde unterschrieben, der Detailprojektplan erstellt und abgenommen und der erste Blögeintrag erstellt. Auf Basis der vorhandenen Informationen/Unterlagen wurde im September 2020 eine erneute Projektplanung durchgeführt.

3.2 Arbeitspaket 2 – IoT Protokolle

Recherche hinsichtlich IoT-Kommunikationsprotokollen auf Layer 1 (PHY) und Layer 2 (MAC) (802.15.4, 6LoWPAN, ZigBee, OpenThread).

Eine Testumgebung bzw. ein Testnetzwerk wurde mit SAMR21-xpro Mikrocontrollern als constrained IoT Devices und einem Raspberry Pi 3B+ als Border Router für die Verbindung zum Internet aufgebaut (siehe u.a. Blogposts und Projekt-Dokumentation). In diesem wurden die Kommunikationsprotokolle IEEE 802.15.4/6LoWPAN und 802.15.4/OpenThread für die Kommunikation zwischen IoT Devices und Border Router getestet.

6LoWPAN und OpenThread (beide basierend auf 802.15.4) wurden im Zuge der Evaluierung der IoT-Drahtloskommunikationsprotokolle zu unseren Favoriten und kamen für die Verwendung in unserem „Proof-of-Concept“ Aufbau in Frage. Die Kommunikation mittels 6LoWPAN konnte mit der Verwendung eines OpenLabs 802.15.4 Radio Moduls am Raspberry Pi erfolgreich durchgeführt werden. Hierbei wurde festgestellt, dass dieses Modul nicht vollständig mit einem Raspberry Pi 4 kompatibel ist. Mit diesem war es nur möglich Pakete mittels 802.15.4 an einen SAMR21-xpro zu senden, aber nicht zu empfangen. Dies konnte gelöst werden, indem stattdessen ein Raspberry Pi 3B+ verwendet wurde. Die Kommunikation mittels OpenThread, unter Verwendung des Nordic nRF52840 Dongle, konnte während der Projektzeit nicht finalisiert werden und wird in der Nachprojektphase weiter bearbeitet. OpenThread wird grundsätzlich von RIOT-OS unterstützt, die Verwendung ist jedoch kaum dokumentiert und es gibt keine öffentlichen Beispiel-Implementierungen.

Mit der Recherche hinsichtlich CoAP-Protokoll wurde in diesem Arbeitspaket begonnen und in Arbeitspaket 4 fortgeführt.

Weiters wurde im Zuge dieses Arbeitspakets auch das Matrix-Protokoll und im Speziellen dessen Client-Server API betrachtet, um die für das „CoMatrix-Protokoll“ relevanten Funktionalitäten zu definieren. Verständnis für die Funktionsweise des Matrix-Protokolls und insbesondere von dessen Client-Server-API ist nun vorhanden.

Blogposts wurden erstellt („IoT Protocols“, „CoMatrix Architecture & Testbed“, „Raspberry Pi & 6LoWPAN“).

Alle Erkenntnisse wurden ausführlich dokumentiert. Für die Kommunikation von Mikrocontrollern via IEEE 802.15.4 via eines Border Routers mit dem Internet gibt wenig öffentlich zugängliche Informationen. Diese Lücke wird mit der durch uns veröffentlichten Dokumentation etwas gefüllt.

Der Betrieb eines Sensornetzwerkes mit Standard Matrix Protokoll wurde als nicht notwendig erachtet, da dies schon öffentlich durch andere Personen demonstriert wurde und wenig Erkenntnisgewinn zu erwarten war.

Aufgrund der Stundenanzahl, die von Herrn Gadermaier in AP4 verbraucht wurde, sind in AP2 auch Tätigkeiten, die eigentlich Teil von AP4 wären, durchgeführt worden.

Durch den Weggang des initialen Projektteams ist es zu Brüchen im Projektfortschritt gekommen.

Ein quantitativer Vergleich mit anderen IoT Protokollen wie z.B. MQTT ist noch ausständig.

3.3 Arbeitspaket 3 – *Matrix Infrastruktur*

Der Matrix Homeserver „Synapse“ wurde für Testzwecke in einer virtuellen Debian Maschine installiert. Anschließend wurde die Installation von „Synapse“ auf einem Raspberry Pi 4 für den Aufbau der Testumgebung bzw. des Testnetzwerkes durchgeführt. Die Installation bzw. das Setup von Synapse wurden einerseits direkt am Host durchgeführt, als auch mittels Docker Image. Als Datenbanken für Synapse wurden Postgres und SQLite getestet.

Verständnis für die Installation, Konfiguration und den Betrieb eines Matrix-Homeservers konnte so gesammelt werden. Die Erkenntnisse wurden ausführlich dokumentiert.

Auf dem Matrix-Homeserver wurden mehrere User angelegt, um die Client-Server API über die Kommandozeile für besseres Verständnis zu testen.

Die Installation bzw. der Betrieb des CoMatrix Gateways wurde im Zuge des Arbeitspakets 4 durchgeführt, da dies mit der Entwicklung desselben Hand-in-Hand ging.

3.4 Arbeitspaket 4 – *Matrix for Constrained Devices*

Recherche und Vergleich verschiedener IoT/Mikrocontroller Betriebssysteme, um ein geeignetes zu wählen. Anforderungen an das Betriebssystem waren die Unterstützung von CoAP, UDP, IPv6, 6LoWPAN, DTLS und OpenThread. RIOT-OS wurde als Betriebssystem ausgewählt, da es kompatibel mit 802.15.4, 6LoWPAN und OpenThread ist und alle anderen Anforderungen erfüllt. Außerdem haben wir in der Lehre (Mobile and Embedded Security Übung) bereits Erfahrung in der Entwicklung von Firmware basierend auf RIOT OS gesammelt.

Das Atmel SAM R21 Xplained Pro (SAMR21-xpro) Board wurde als Hardwareplattform gewählt, da dieser Mikrocontroller als Standard in den RIOT-OS Libraries verwendet wird und bereits Erfahrungswerte im Projektteam vorhanden sind.

Die notwendige Hardware wurde bestellt:

- 5x Raspberry Pi 4B Bundle
- 5x Raspberry Pi - Adapter micro-HDMI auf HDMI (1m)

- 5x DS18S20+ DIGITAL THERMOMETER
- 5x Atmel SAM R21 Xplained Pro (SAMR21-xpro) Board
- 5x Nordic NRF52840-DONGLE

Die Auswahl geeigneter Constrained Networks wurde im AP2 bearbeitet. Ebenso die Kommunikation von Constrained Node zu Gateway per Constrained Stack.

Es wurde eine Recherche und Evaluierung hinsichtlich bestehender CoAP-http Cross-Protocol Proxies durchgeführt (aiocoap, IBM crosscoap, CoAPthon). Diese waren jedoch aus unterschiedlichen Gründen nicht für CoMatrix geeignet bzw. nicht flexibel genug. Das Testen dieser Proxies führte zu einem tiefergehenden Verständnis des CoAP Protokoll.

Daraufhin wurden unterschiedliche CoAP Programmbibliotheken verglichen (aiocoap, CoAPthon, CoAPthon3, Eclipse Californium, libcoap). Python/aiocoap wurde als CoAP-Programmbibliothek gewählt, da es einfache API, eine ausführliche Dokumentation und diverse Beispielimplementierungen zur Verfügung stellt. Außerdem eignete sich aiocoap gut, um die Kommunikation über 6LoWPAN mit bestehenden RIOT-OS Examples zu testen (gnrc_networking, gcoap) und somit zeitgleich die CoMatrix Library für RIOT-OS zu entwickeln. Aiocoap bietet leider keine Server-seitige Unterstützung von DTLS. Die Unterstützung von DTLS wird in der Nachprojektphase nachgeliefert, um Transport Security zu CoMatrix hinzuzufügen. Dafür wird eventuell die DTLS-Unterstützung in aiocoap ergänzt, ansonsten bliebe nur Re-Implementierung mit einer CoAP-Programmbibliothek, die DTLS defaultmäßig unterstützt (z.B. Eclipse Californium oder libcoap). Die Installation bzw. der Betrieb des CoMatrix Gateways wurde im Zuge des AP4 durchgeführt, da dies mit der Entwicklung desselben Hand-in-Hand ging.

Außerdem wurde die CoMatrix Library als externes Software-Modul für RIOT-OS entwickelt. Zwei Beispiel-Anwendungen wurden mithilfe dieser Bibliothek implementiert (CoMatrix chat und CoMatrix tempsensor). Zum einen ein CoMatrix Commandline Chat Client, der zum Test aller implementierten Matrix Funktionalität verwendet werden kann. Die Anwendung ermöglicht die Registrierung eines neuen Matrix Benutzers, das An- und Abmelden eines Benutzers und das Hinzufügen eines Benutzers zu einem Matrix Raum. Sowie das Senden von Nachrichten in einen Raum und das Empfangen der zuletzt gesendeten Nachricht eines Matrix Raums.

Im zweiten implementierten Anwendungsfall veranschaulicht ein Temperatur Sensor Knoten, der kontinuierlich gemessene Temperaturwerte in einen Matrix Raum sendet, die Verwendung der CoMatrix Client Bibliothek.

3.5 Arbeitspaket 5 – Öffentlichkeitsarbeit

Die Haupttätigkeiten dieses APs war die Bekanntmachung von CoMatrix und das Schreiben von Blogposts.

Das Projekt wurde im Zuge eines Vortrags von Mathias Tausig auf den „Linuxwochen Wien“ am 05.07.2019 präsentiert. Der Vortrag wurde von ca. 40 Personen besucht.

Diverse Blogposts auf der Netidee-Projektwebsite berichten über den Fortschritt des Projektes.

Die Domain „comatrix.eu“ wurde angeschafft und die Website mit Inhalt gefüllt.

Leider wurden in den Jahren 2020 und 2021 viele Konferenzen, Tagungen, etc. aufgrund der Covid19-Pandemie abgesagt oder finden nur online statt.

3.6 Arbeitspaket 6 – *Wissenschaftliche Verarbeitung*

Seitens der Matrix Community gibt es seit längeren Bestrebungen das Matrix Protokoll für IoT Geräte nutzbar zu machen (siehe <https://matrix.org/iot/>), dabei können wir unsere Ergebnisse sinnvoll einbringen. Der Grundstein für die wissenschaftliche Verarbeitung ist mit der Implementierung gelegt. An der Verschlüsselung der Kommunikation zwischen Gateway und IoT Node werden wir noch weiterarbeiten. Aus aktueller Sicht kommen dafür Verschlüsselungstechniken auf Transportebene mit DTLS, auf Applikationsebene mit OSCORE oder die Verwendung des OpenThread Netzwerkstacks in Frage. Es ist geplant diese Verfahren beispielweise hinsichtlich der Performance (Laufzeit, Memoryfootprint des Firmwarecodes, Kommunikations-Bandbreite), als auch der Unterstützung durch unterschiedliche Plattformen zu evaluieren und wissenschaftlich aufbereiten.

Außerdem arbeiten wir an der Einreichung eines Papers bei einer Konferenz, die sich mit dem Thema IoT auseinandersetzt (z. B. Konferenz IOTBDS 2022).

3.7 Arbeitspaket 7 – *Projektmanagement*

Laufende Projektplanung und Abstimmung des Projektteams.

Zwischenbericht wurde erstellt. Stundenlisten wurden erstellt und wurden administriert.

FH interner Projektfortschrittsbericht wurde erstellt.

Kleine Unklarheiten hinsichtlich der Vorgangsweise für den Zwischenbericht aufgrund der langen Verzögerung des Projektes.

Erhöhter Abstimmungsbedarf im Projektteam aufgrund der Covid19-Situation (Home Office etc.).

3.8 Arbeitspaket N – *Dokumentation und Formales am Projektende*

Der Projektverlauf wurde im Projektendbericht, in der Zusammenfassung und der Endabrechnung dokumentiert.

Alle notwendigen Dokumenten wurde an Netidee übermittelt und auf die Projektwebsite hochgeladen.

Ein abschließender Blogbeitrag wurde verfasst.

4 Umsetzung Förderauflagen

In der Fördervereinbarung wurden keine speziellen Förderauflagen festgelegt.

5 Liste Projektergebnisse

1	<i>Projektzwischenbericht</i>	CC-BY-SA	https://www.netidee.at/comatrix
2	<i>Projektendbericht</i>	CC-BY-SA	https://www.netidee.at/comatrix
3	<i>Entwickler*innen-DOKUMENTATION</i>	CC-BY-SA	https://comatrix.eu/
4	<i>Anwender*innen-DOKUMENTATION</i>	CC-BY-SA	https://comatrix.eu/
5	<i>Einseiter/Zusammenfassung</i>	CC-BY-SA	https://www.netidee.at/comatrix
6	<i>Präsentation Linuxwochen 2019</i>	CC-BY-SA	https://www.netidee.at/comatrix
7	<i>CoMatrix Gateway</i>	GNU GPL v3	https://gitlab.com/comatrix
8	<i>CoMatrix Client Software/Library</i>	GNU GPL v3	https://gitlab.com/comatrix

6 Verwertung der Projektergebnisse in der Praxis

Eine konkrete Verwendung in einem praktischen Umfeld konnte während der Projektlaufzeit noch nicht erreicht werden.

Wir sind jedenfalls gespannt auf die Rückmeldungen aus der Community nach der Veröffentlichung über „This Week in Matrix“ (siehe Öffentlichkeitsarbeit/Vernetzung).

Das Projekt wird im Rahmen der Lehr- und Forschungstätigkeit an der FH Campus Wien weitergeführt und auch über diese finanziert. Die aus CoMatrix resultierenden Projektergebnisse und Erkenntnisse können den Studierenden in Lehrveranstaltungen vermittelt werden (u.a. LV „Mobile and Embedded Security“). Auch eine Weiterentwicklung des Projektes u.a. im Rahmen von Projektarbeiten sowie Abschlussarbeiten ist geplant. Daraus ergibt sich in jedem Fall zumindest eine regelmäßige Aktualisierung der Projektergebnisse.

7 Öffentlichkeitsarbeit/ Vernetzung

Am 05.07.2019 erfolgte eine erste Präsentation von CoMatrix durch Mathias Tausig im Zuge der Linuxwochen Wien. Der Vortrag wurde von ca. 40 Personen besucht.

Nach Projektabschluss wird eine Bekanntmachung von CoMatrix über „This Week in Matrix (TWIM)“ vorgenommen (siehe <https://matrix.org/blog/category/this-week-in-matrix>).

In der Nachprojektphase wollen wir CoMatrix beim RIOT-OS Summit (<https://summit.riot-os.org/2021/>) im September 2021 der RIOT-OS Community präsentieren. Dieses findet von 9.-10. September statt (Online Event).

Weiters ist die Einreichung beim Chaos Communication Congress 2021 und bei der FOSDEM 2022 (Matrix Community) angedacht. Außerdem arbeiten wir an der Einreichung eines wissenschaftlichen Papers bei einer Konferenz, die sich mit dem Thema IoT auseinandersetzt (z. B. Konferenz IOTBDS 2022).

Wir werden die allgemein verwendbaren Dokumentation (z.B. Raspberry Pi und 6LoWPAN) auch auf <https://wiki.elvis.science> veröffentlichen.

Die Projekt-Website ist unter der Domain „comatrix.eu“ erreichbar.

8 Eigene Projektwebsite

Die Projekt-Website ist unter der Domain „comatrix.eu“ erreichbar (<https://comatrix.eu/>).

9 Geplante Aktivitäten nach netidee-Projektende

Die Implementierung von Verschlüsselung bzw. Transport Security mittels DTLS/OSCORE/OpenThread ist geplant (siehe Abschnitt „3.6 Arbeitspaket 6 – Wissenschaftliche Verarbeitung“).

Die clientseitige CoMatrix Library könnte um „CoAP block-wise transfers“ (RFC7959) ergänzt werden.

Die Integration des CoMatrix-Moduls als Standard-Modul im RIOT-OS wird angestrebt.

Wir arbeiten an der Einreichung eines wissenschaftlichen Papers bei einer Konferenz, die sich mit dem Thema IoT auseinandersetzt (z. B. Konferenz IOTBDS 2022).

Siehe Abschnitte „6 Verwertung der Projektergebnisse in der Praxis“ und „7 Öffentlichkeitsarbeit/Vernetzung“.

10 Anregungen für Weiterentwicklungen durch Dritte

Diverse Weiterentwicklungen, z.B. mehr Funktionalität am Gateway wie die Verwaltung der Homeserver-Daten am Gateway (statt über den Client) oder das Hinzufügen von zusätzlichen Matrix Client-Server API Funktionalitäten.

Testen von CoMatrix auf anderen (RIOT-OS kompatiblen) Mikrocontroller Boards.

Testen von CoMatrix in einem 802.15.4 Mesh-Netzwerk.

Testen von CoMatrix mit einem anderen IoT-Kommunikationsprotokoll als 802.15.4/6LoWPAN.

Portierung der clientseitigen CoMatrix Library für andere IoT-Betriebssysteme (z.B. Contiki).

Einsatz in diversen Smart-Home Projekten.